

A note on the efficient implementation of Hamiltonian BVMs

Luigi Brugnano^a, Felice Iavernaro^b, Donato Trigiante^c

^a*Dipartimento di Matematica, Università di Firenze, Viale Morgagni 67/A, 50134 Firenze (Italy).*

^b*Dipartimento di Matematica, Università di Bari, Via Orabona 4, 70125 Bari (Italy).*

^c*Dipartimento di Energetica, Università di Firenze, Via Lombroso 6/17, 50134 Firenze (Italy).*

Abstract

We discuss the efficient implementation of Hamiltonian BVMs (HBVMs), a recently introduced class of energy preserving methods for canonical Hamiltonian systems (see [2] and references therein), via their *blended* formulation. We also discuss the case of separable problems, for which the structure of the problem can be exploited to gain efficiency.

Keywords: Ordinary differential equations, Runge-Kutta methods, one-step methods, Hamiltonian problems, separable problems, Hamiltonian Boundary Value Methods, energy preserving methods, blended methods, symplectic methods, energy drift

2000 MSC: 65P10, 65L05

1. Introduction

The conservation of energy allows to avoid the numerical drift observed when using standard numerical methods for solving canonical Hamiltonian problems, i.e., problems in the form

$$y' = J\nabla H(y), \quad J = \begin{pmatrix} 0 & I_m \\ -I_m & 0 \end{pmatrix}, \quad y(t_0) = y_0 \in \mathbb{R}^{2m}, \quad (1)$$

where $H(y)$ is a smooth scalar function and, in general, I_r will hereafter denote the identity matrix of dimension r (when the lower index will be omitted, the size of the matrix can be deduced from the context). In this respect, *Hamiltonian Boundary Value Methods (HBVMs)* is a recently introduced class of methods able to conserve energy when $H(y)$ is a polynomial of arbitrarily high degree. Clearly, this implies a *practical* conservation of energy for any suitably regular Hamiltonian function, which will be assumed hereafter. We refer to [2, 3, 4, 5, 6, 7], and references therein, for an overview on energy-conserving methods and the derivation of HBVMs. When problem (1) is separable, i.e., when

$$H(y) \equiv H(q, p) = \frac{1}{2}p^T p - U(q), \quad q, p \in \mathbb{R}^m, \quad (2)$$

^{*}Work developed within the project “Numerical methods and software for differential equations”.

Email addresses: luigi.brugnano@unifi.it (Luigi Brugnano), felix@dm.uniba.it (Felice Iavernaro), trigiant@unifi.it (Donato Trigiante)

then (1) reduces to a special second order equation,

$$q'' = \nabla U(q),$$

and the associated HBVM may be properly formulated in order to take advantage, in terms of efficiency, from the above simplification.

In this paper we investigate the efficient implementation of HBVMs, also in the case of separable problems. In more details, in Section 2 we briefly derive HBVMs. Then, in Section 3 we investigate the efficient solution of the generated discrete problem, via the *blended* implementation of the methods, which has already proved to be very effective in other settings (see, e.g., [1, 8, 9, 10, 11, 12, 13, 14]). The case of separable problems is then discussed in Section 4. A few numerical tests, along with some concluding remarks are then given in Section 5.

2. Hamiltonian BVMs (HBVMs)

The derivation of HBVMs will be done according to the approach followed in [6, 7], which further simplifies the already simple idea initially used in [2, 3, 4, 5] (see also [15, 16]). Let us then consider the restriction of problem (1) to the interval $[t_0, t_0 + h]$, with the right-hand side expanded along an orthonormal basis $\{\hat{P}_j\}_{j \geq 0}$:

$$y'(t_0 + \tau h) = J \sum_{j \geq 0} \hat{P}_j(\tau) \int_0^1 \hat{P}_j(c) \nabla H(y(t_0 + ch)) dc, \quad \tau \in [0, 1]. \quad (3)$$

In particular, we here consider an orthonormal polynomial basis, provided by the shifted and scaled Legendre polynomials on the interval $[0, 1]$, even though the arguments can be easily extended to more general bases. The basic idea, is now that of looking for an approximate solution belonging to the set of polynomials of degree not larger than s . This is achieved by truncating the series at the right-hand side in (3), thus obtaining the approximate problem

$$\sigma'(t_0 + \tau h) = J \sum_{j=0}^{s-1} \hat{P}_j(\tau) \int_0^1 \hat{P}_j(c) \nabla H(\sigma(t_0 + ch)) dc, \quad \tau \in [0, 1], \quad \sigma(t_0) = y_0. \quad (4)$$

The approximation to $y(t_0 + h)$ is then given by

$$y_1 \equiv \sigma(t_0 + h). \quad (5)$$

The method can be easily seen to be energy-preserving since, considering that J is skew-symmetric,

$$\begin{aligned} H(y_1) - H(y_0) &= h \int_0^1 \nabla H(\sigma(t_0 + \tau h))^T \sigma'(t_0 + \tau h) d\tau \\ &= h \sum_{j=0}^{s-1} \left[\int_0^1 \hat{P}_j(\tau) \nabla H(\sigma(t_0 + \tau h)) d\tau \right]^T J \left[\int_0^1 \hat{P}_j(c) \nabla H(\sigma(t_0 + ch)) dc \right] = 0. \end{aligned}$$

Integrating both sides of the first equation in (4) yields

$$\sigma(t_0 + \tau h) = y_0 + h \sum_{j=0}^{s-1} \int_0^\tau \hat{P}_j(x) dx \int_0^1 \hat{P}_j(c) J \nabla H(\sigma(t_0 + ch)) dc, \quad (6)$$

which may be exploited to determine the shape of the unknown polynomial σ , provided that a technique to handle the rightmost integrals is taken into account: the obvious choice is the use of quadrature formulae. If we assume that $H(y)$ is a polynomial of degree ν , then the integrals appearing in (4) can be exactly computed by a Gaussian formula with k abscissas $\{c_i\}$, in the event that

$$k \geq s\nu/2, \quad (7)$$

thus obtaining a discrete problem in the form

$$\sigma(t_0 + c_i h) \equiv \sigma_i = y_0 + h \sum_{j=0}^{s-1} \int_0^{c_i} \hat{P}_j(x) dx \sum_{\ell=1}^k b_\ell \hat{P}_j(c_\ell) J \nabla H(\sigma_\ell), \quad i = 1, \dots, k, \quad (8)$$

where the $\{b_i\}$ are the quadrature weights of the formula defined over the abscissae $\{c_i\}$. For general, suitably regular (e.g., analytical) Hamiltonian functions, we can still use formula (8) in place of (6), provided that the integrals in (6) are approximated to machine precision¹: in the following, we will always assume such an accuracy level when a non polynomial function is considered, and consequently we will make no distinction between the integrals and the corresponding approximations as well as between the two polynomials σ obtained by solving either (8) or (6) (see [7] for more details).

Method (8)-(5) is called HBVM(k, s): it was shown [4] that its order is $2s$, for all $k \geq s$. In particular, for $k = s$ it reduces to the well known s -stages Gauss method.

By introducing the matrices $\Omega = \text{diag}(b_1, \dots, b_k)$ and

$$\mathcal{I}_{s-1} = \left(\int_0^{c_i} \hat{P}_{j-1}(x) dx \right)_{\substack{i=1 \dots k \\ j=1 \dots s}} \in \mathbb{R}^{k \times s}, \quad \mathcal{P}_{s-1} = \left(\hat{P}_{j-1}(c_i) \right)_{\substack{i=1 \dots k \\ j=1 \dots s}} \in \mathbb{R}^{k \times s},$$

the HBVM(k, s) can be recast as a Runge-Kutta method with Butcher tableau

$$\begin{array}{c|c} c_1 & \\ \vdots & \\ c_k & \\ \hline & b_1 \quad \dots \quad b_k \end{array} \quad A \equiv \mathcal{I}_{s-1} \mathcal{P}_{s-1}^T \Omega \quad (9)$$

The next result follows from well-known properties of Legendre polynomials (hereafter e_i denotes the i th unit vector in \mathbb{R}^s).

Lemma 1.

$$\mathcal{I}_{s-1} = \mathcal{P}_s \hat{X}_s \equiv \mathcal{P}_s \begin{pmatrix} X_s \\ \xi_s e_s^T \end{pmatrix}, \quad (10)$$

where

$$X_s = \begin{pmatrix} \frac{1}{2} & -\xi_1 & & \\ \xi_1 & 0 & \ddots & \\ & \ddots & \ddots & -\xi_{s-1} \\ & & \xi_{s-1} & 0 \end{pmatrix}, \quad \xi_i = \frac{1}{2\sqrt{4i^2 - 1}}, \quad i \geq 1. \quad (11)$$

¹As we will see, increasing the order of the quadrature formula, namely the integer k , will not result into an increase of the computational cost associated with the implementation of the method.

Consequently, the matrix in the Butcher tableau (9) can be written as

$$A = \mathcal{P}_s \hat{X}_s \mathcal{P}_{s-1}^T \Omega. \quad (12)$$

Notice that, since $\mathcal{P}_s \hat{X}_s$ has s linearly independent columns, the $k \times k$ coefficient matrix A has rank s : it is then possible to recast the discrete problem in a more convenient form, which clearly shows that its (block) size is s , rather than k (see also [3]). For this purpose, let us define the (block) vectors (see (4) and (8))

$$\mathbf{y} = \begin{pmatrix} \sigma_1 \\ \vdots \\ \sigma_k \end{pmatrix}, \quad \boldsymbol{\gamma} = \begin{pmatrix} \gamma_0 \\ \vdots \\ \gamma_{s-1} \end{pmatrix}, \quad \gamma_j = \sum_{\ell=1}^k b_\ell \hat{P}_j(c_\ell) J \nabla H(\sigma(t_0 + c_\ell h)), \quad j = 0, \dots, s-1. \quad (13)$$

In view of (4), we see that the vectors γ_j may be interpreted as the coefficients in the expansion of the degree $s-1$ polynomial $\sigma'(t_0 + \tau h)$ along the orthonormal basis $\{\hat{P}_j\}_{j=0, \dots, s-1}$.

From (8) one obtains

$$\mathbf{y} = e \otimes y_0 + h \mathcal{I}_{s-1} \otimes I_{2m} \boldsymbol{\gamma}, \quad (14)$$

with $e = (1, \dots, 1)^T \in \mathbb{R}^k$, and then, by virtue of (13), one has to solve the equation in the unknown $\boldsymbol{\gamma}$

$$F(\boldsymbol{\gamma}) \equiv \boldsymbol{\gamma} - (\mathcal{P}_{s-1}^T \Omega \otimes J) \nabla H(e \otimes y_0 + h \mathcal{I}_{s-1} \otimes I_{2m} \boldsymbol{\gamma}) = 0. \quad (15)$$

The application of the simplified Newton iteration for solving (15) yields

$$[I - h \mathcal{P}_{s-1}^T \Omega \mathcal{I}_{s-1} \otimes G_0] \Delta^\ell = -F(\boldsymbol{\gamma}^\ell), \quad \boldsymbol{\gamma}^{\ell+1} = \boldsymbol{\gamma}^\ell + \Delta^\ell, \quad (16)$$

with $G_0 = (J \nabla^2 H(y_0))$. By virtue of (10), and considering that

$$\mathcal{P}_{s-1}^T \Omega \mathcal{P}_s = (I_s \ 0) \in \mathbb{R}^{s \times s+1}, \quad (17)$$

(16) reduces to

$$[I - h X_s \otimes G_0] \Delta^\ell = -F(\boldsymbol{\gamma}^\ell), \quad \boldsymbol{\gamma}^{\ell+1} = \boldsymbol{\gamma}^\ell + \Delta^\ell, \quad \ell = 0, 1, \dots, \quad (18)$$

which, as is readily seen, has (block) size s , rather than k .

3. Blended implementation

From the arguments in the previous section, one then concludes that the discrete problem, to be solved at each integration step when approximating the Hamiltonian problem (1), is given by (15), thus requiring the solution of (18). We are going to solve such equation by means of a *blended* implementation of the method, according to [1, 8, 9, 14]. Indeed, such implementation of block implicit methods has proved to be very effective, leading to the development of the codes **BiM** [9] and **BiMD** [13] for stiff ODE IVPs and linearly implicit DAEs (the codes are available at the url [17]). Let us, for sake of simplicity, discard the iteration index. Consequently, we have to solve the linear system

$$(I - h X_s \otimes G_0) \Delta = -F(\boldsymbol{\gamma}) \equiv \boldsymbol{\eta}. \quad (19)$$

Considering that matrix X_s (see (11)) is nonsingular, such equation can be equivalently written as

$$\rho (X_s^{-1} \otimes I_{2m} - hI_s \otimes G_0) \Delta = \rho X_s^{-1} \otimes I_{2m} \boldsymbol{\eta} \equiv \boldsymbol{\eta}_1, \quad (20)$$

where ρ is a positive constant. By introducing the (matrix) weight function

$$\theta = I_s \otimes \Sigma_0^{-1}, \quad \Sigma_0 = (I_{2m} - \rho h G_0)^{-1}, \quad (21)$$

we then obtain the following problem, which has still the same solution as (19):

$$T(\Delta) \equiv \theta [(I - hX_s \otimes G_0) \Delta - \boldsymbol{\eta}] + (I - \theta) [\rho (X_s^{-1} \otimes I_{2m} - hI_s \otimes G_0) \Delta - \boldsymbol{\eta}_1] = 0. \quad (22)$$

One easily realizes that it is obtained as the *blending*, with weights θ and $(I - \theta)$, of the two equivalent problems (19) and (20), respectively. Problem (22) defines the *blended method* associated with the original one, which we call *blended HBVM*, in the present case. The free parameter ρ is chosen in order to optimize the convergence properties of the corresponding *blended iteration*,

$$\Delta_{n+1} = \Delta_n - \theta T(\Delta_n), \quad n \geq 0, \quad (23)$$

with an obvious meaning of the lower index. Such iteration only requires (see (21)) the factorization of the matrix Σ_0 having the same size as that of the continuous problem. According to the linear analysis of convergence in [11], the free parameter ρ is chosen as

$$\rho = \rho_s \equiv \min\{|\lambda| : \lambda \in \sigma(X_s)\}, \quad (24)$$

which provides optimal convergence properties (in particular, an L -convergent iteration [11]). A few values of (24) are listed in the table below, for sake of completeness.

| s | 2 | 3 | 4 | 5 |
|----------|--------|--------|--------|--------|
| ρ_s | 0.2887 | 0.1967 | 0.1475 | 0.1173 |

Remark 1. A nonlinear version of (23) can be readily derived, by taking $\Delta_n = 0$ and updating the vectors $\boldsymbol{\eta}$ and $\boldsymbol{\eta}_1$ in (22) at each iteration.

4. The case of separable problems

Let us now apply the method to the separable problem (2). By setting the (block) vectors

$$\mathbf{q} = (q_1^T, \dots, q_k^T)^T, \quad \mathbf{p} = (p_1^T, \dots, p_k^T)^T,$$

one then obtains (see (12)),

$$\mathbf{q} = e \otimes q_0 + hA \otimes I_m \mathbf{p}, \quad \mathbf{p} = e \otimes p_0 + hA \otimes I_m \nabla U(\mathbf{q}),$$

i.e., since $Ae = c \equiv (c_1, \dots, c_k)^T$,

$$\mathbf{q} = e \otimes q_0 + hc \otimes p_0 + h^2 A^2 \otimes I_m \nabla U(\mathbf{q}). \quad (25)$$

Moreover, taking into account (9)–(12) and (17), one obtains

$$A^2 = \mathcal{I}_{s-1} X_s \mathcal{P}_{s-1}^T \Omega. \quad (26)$$

The new approximations to $q(t_0 + h)$ and $p(t_0 + h)$ are then given by

$$q_0 + hp_0 + h^2 e^T \Omega A \otimes I_m \nabla U(\mathbf{q}), \quad p_0 + he^T \Omega \otimes I_m \nabla U(\mathbf{q}),$$

respectively. By using similar arguments as those given in Section 2 (see (14)), we set

$$\mathbf{q} = e \otimes q_0 + hc \otimes p_0 + h^2 \mathcal{I}_{s-1} X_s \otimes I_m \gamma,$$

thus obtaining the following equation (which is the analogous of (15)):

$$F(\gamma) \equiv \gamma - (\mathcal{P}_{s-1}^T \Omega \otimes I_m) \nabla U(e \otimes q_0 + hc \otimes p_0 + h^2 \mathcal{I}_{s-1} X_s \otimes I_m \gamma) = 0. \quad (27)$$

Similarly as what seen in Section 3, the application of the simplified Newton iteration for solving (27) then gives, by virtue of (10) and (17), and setting $G_0 = \nabla^2 U(q_0)$,

$$[I - h^2 X_s^2 \otimes G_0] \Delta^\ell = -F(\gamma^\ell), \quad \gamma^{\ell+1} = \gamma^\ell + \Delta^\ell, \quad \ell = 0, 1, \dots, \quad (28)$$

which, as in the previous case, has (block) size s , rather than k . The problem is then exactly that seen in (18), via the formal substitutions

$$h \longrightarrow h^2, \quad X_s \longrightarrow X_s^2. \quad (29)$$

This means that we can repeat similar steps for the *blended* solution of (28), by following the same arguments seen in Section 3. In more details, (19)–(23) can be repeated, by considering the formal substitutions (29) and, moreover,

$$\rho \longrightarrow \rho^2, \quad I_{2m} \longrightarrow I_m.$$

Also in this case [10, 11], the optimal choice of the parameter ρ turns out to be given by (24).

5. Numerical Tests

We here consider a model problem to test the proposed algorithms and methods, in order to confirm the usefulness of the proposed approach. In particular, it is clear that a Newton-type iteration, like (18) and (28), works well when the linear part of the problem is significant. For this purpose, we consider the following polynomial Hamiltonian,

$$H(q, p) = \frac{1}{2}p^2 - 10^4 q^2 \left(\frac{4}{5}q^3 - \frac{3}{4}q^2 - \frac{2}{3}q + \frac{1}{2} \right), \quad (30)$$

from which we derive the following special second order problem,

$$q'' = 10^4 q (4q^3 - 3q^2 - 2q + 1), \quad t \in [0, 100], \quad q(0) = 0, \quad q'(0) = 1. \quad (31)$$

For solving (31), we use the following fourth-order numerical methods:

- the symplectic 2-stages Gauss method (GAUSS2);
- HBVM(8,2) which is energy conserving, for the problem at hand.

Table 1: total number of iterations for solving the discrete problems with the specified stepsize h (– if no convergence).

| h | GAUSS2 | | | | HBVM(8,2) | | | |
|-------------------|--------------|-------------|-------------|-------------|--------------|-------------|-------------|-------------|
| | second order | | first order | | second order | | first order | |
| | blended | fixed-point | blended | fixed-point | blended | fixed-point | blended | fixed-point |
| 10^{-3} | 664545 | 690197 | 952902 | 1217673 | 660317 | 695765 | 947618 | 1225318 |
| $5 \cdot 10^{-3}$ | 242844 | – | 308406 | – | 228242 | 223883 | 293949 | 424402 |
| 10^{-2} | – | – | – | – | 194163 | – | 253049 | – |

For both methods, we consider a fixed-step implementation with stepsize h , with the generated discrete problems solved either with a fixed-point iteration or with a blended iteration, which have approximately the same cost, in terms of function evaluations. Moreover, we also compare the second order implementation described in Section 4, with the equivalent first order Hamiltonian formulation of the problem, as described in Section 3. Table 1 summarizes the obtained results, in terms of total number of iterations (blended or fixed-point) for covering the specified integration interval. From the listed results, one deduces that the second order formulation of the problem is less demanding in terms of needed iterations. Moreover, the blended iteration turns out to be both more efficient and robust than the fixed-point iteration.

Finally, in Figures 1–4 we plot the phase portraits of the numerical solutions, for the two methods and the various stepsizes, along with the corresponding error in the numerical Hamiltonian. As one can see, the phase portraits of the HBVM(8,2) method are always correct, whatever the used stepsize (see Figure 1 and the left plot in Figure 2), since the Hamiltonian is conserved (up to round-off errors), as is shown in the right plot of Figure 2. This is not true for the GAUSS2 method, for which the error in the Hamiltonian depends on the used stepsize, as is shown in Figure 4, thus causing drawbacks in the corresponding phase portraits of the numerical solution, unless the stepsize is very small (see the two plots of Figure 3).

From the numerical tests, one can then conclude that the proposed blended implementation of HBVMs turns out to be robust and efficient. Moreover, the energy-conserving property of such methods turns out to be very remarkable, with respect to standard symplectic methods. Finally, the second order formulation of HBVMs greatly improves their performance.

References

- [1] L.Brugnano. Blended Block BVMs (B₃VMs): A Family of Economical Implicit Methods for ODEs, *Journal of Computational and Applied Mathematics* **116** (2000) 41–62.
- [2] L.Brugnano, F.Iavernaro and D.Trigiant. The Hamiltonian BVMs (HBVMs) Homepage, [arXiv:1002.2757](http://www.math.unifi.it/~brugnano/HBVM/) (URL: <http://www.math.unifi.it/~brugnano/HBVM/>).
- [3] L.Brugnano, F.Iavernaro and D.Trigiant. Analysis of Hamiltonian Boundary Value Methods (HBVMs): a class of energy-preserving Runge-Kutta methods for the numerical solution of polynomial Hamiltonian dynamical systems, (2009) (submitted) ([arXiv:0909.5659](https://arxiv.org/abs/0909.5659)).
- [4] L.Brugnano, F.Iavernaro and D.Trigiant. Hamiltonian Boundary Value Methods (Energy Preserving Discrete Line Integral Methods), *Jour. of Numer. Anal. Industr. and Appl. Math.* **5**,1–2 (2010) 17–37. ([arXiv:0910.3621](https://arxiv.org/abs/0910.3621))

- [5] L.Brugnano, F.Iavernaro and D.Trigiane. Isospectral Property of Hamiltonian Boundary Value Methods (HBVMs) and their connections with Runge-Kutta collocation methods, *Preprint* (2010) ([arXiv:1002.4394](#)).
- [6] L.Brugnano, F.Iavernaro and D.Trigiane. Numerical Solution of ODEs and the Columbus' Egg: Three Simple Ideas for Three Difficult Problems. *Mathematics in Engineering, Science and Aerospace* **1**,4 (2010) 105–124. ([arXiv:1008.4789](#))
- [7] L.Brugnano, F. Iavernaro and D. Trigiane. A unifying framework for the derivation and analysis of effective classes of one-step methods for ODEs. *Preprint* (2010) ([arXiv:1009.3165](#)).
- [8] L.Brugnano and C.Magherini. Blended Implementation of Block Implicit Methods for ODEs, *Appl. Numer. Math.* **42** (2002) 29–45.
- [9] L.Brugnano and C. Magherini. The BiM Code for the Numerical Solution of ODEs, *Jour. Comput. Appl. Mathematics* **164–165** (2004) 145–158.
- [10] L.Brugnano and C. Magherini. Blended Implicit Methods for solving ODE and DAE problems, and their extension for second order problems, *Jour. Comput. Appl. Mathematics* **205** (2007) 777–790.
- [11] L.Brugnano and C.Magherini. Recent Advances in Linear Analysis of Convergence for Splittings for Solving ODE problems, *Applied Numerical Mathematics* **59** (2009) 542–557.
- [12] L.Brugnano and C.Magherini. Blended General Linear Methods based on Boundary Value Methods in the GBDF family, *Jour. of Numer. Anal., Ind. and Appl. Math.* **4**,1–2 (2009) 23–40.
- [13] L.Brugnano, C. Magherini, and F. Mugnai. Blended Implicit Methods for the Numerical Solution of DAE Problems, *Jour. Comput. Appl. Mathematics* **189** (2006) 34–50.
- [14] L.Brugnano and D. Trigiane. Block Implicit Methods for ODEs, in *Recent Trends in Numerical Analysis*, D.Trigiane ed., Nova Science Publ. Inc., New York, 2001, pp. 81–105.
- [15] F. Iavernaro and B. Pace, Conservative Block-Boundary Value Methods for the solution of Polynomial Hamiltonian Systems, *AIP Conf. Proc.* **1048** (2008) 888–891.
- [16] F. Iavernaro and D. Trigiane. High-order Symmetric Schemes for the Energy Conservation of Polynomial Hamiltonian Problems, *Jour. of Numer. Anal., Ind. and Appl. Math.* **4**,1–2 (2009) 87–101.
- [17] *The codes BiM and BiMD Home Page:* <http://www.math.unifi.it/~brugnano/BiM/>

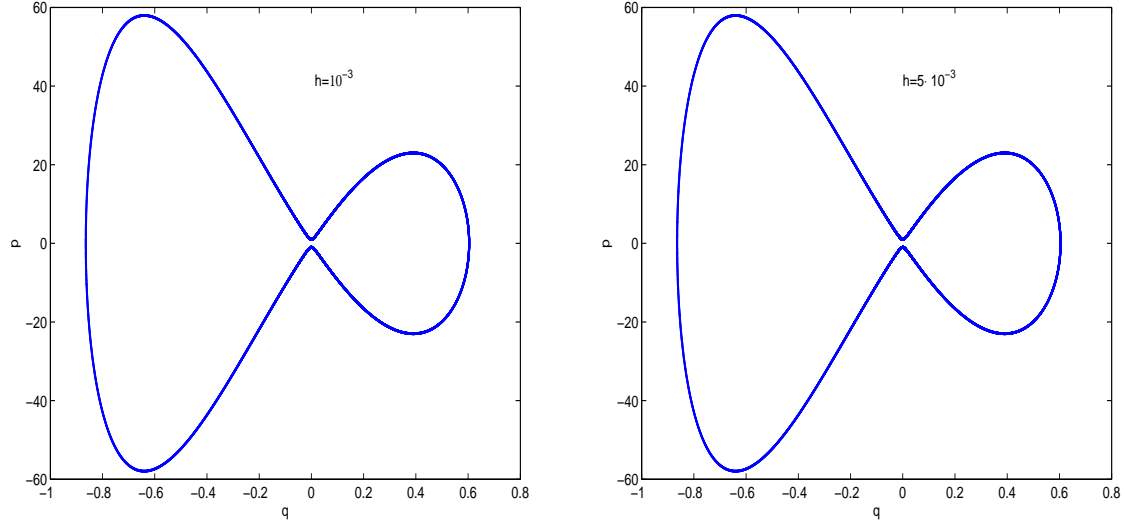


Figure 1: phase portraits for HBVM(8,2), $h = 10^{-3}$ (left) and $h = 5 \cdot 10^{-3}$ (right).

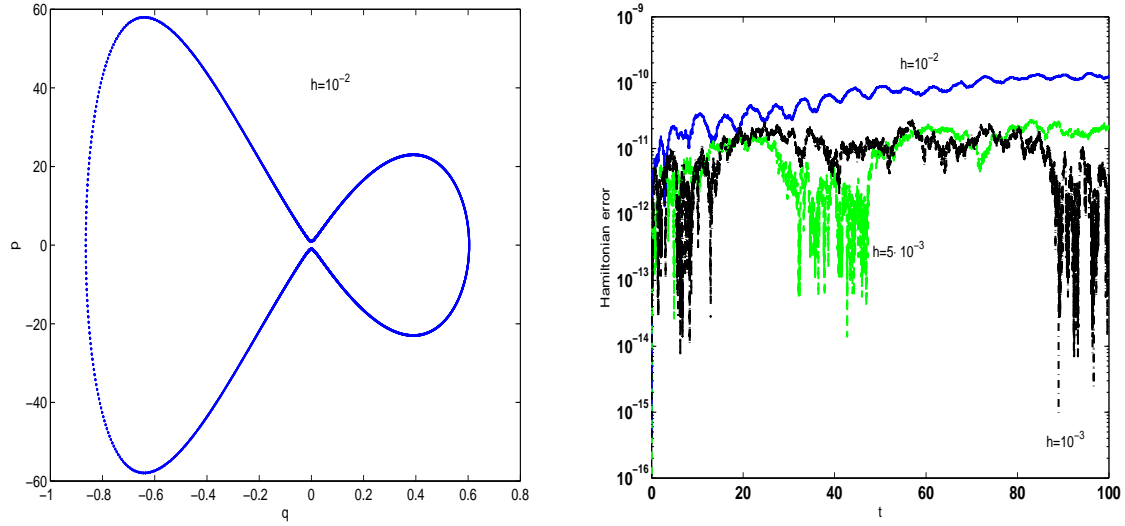


Figure 2: phase portrait for HBVM(8,2), $h = 10^{-2}$ (left) and Hamiltonian error $h = 10^{-3}, 5 \cdot 10^{-3}, 10^{-2}$ (right).

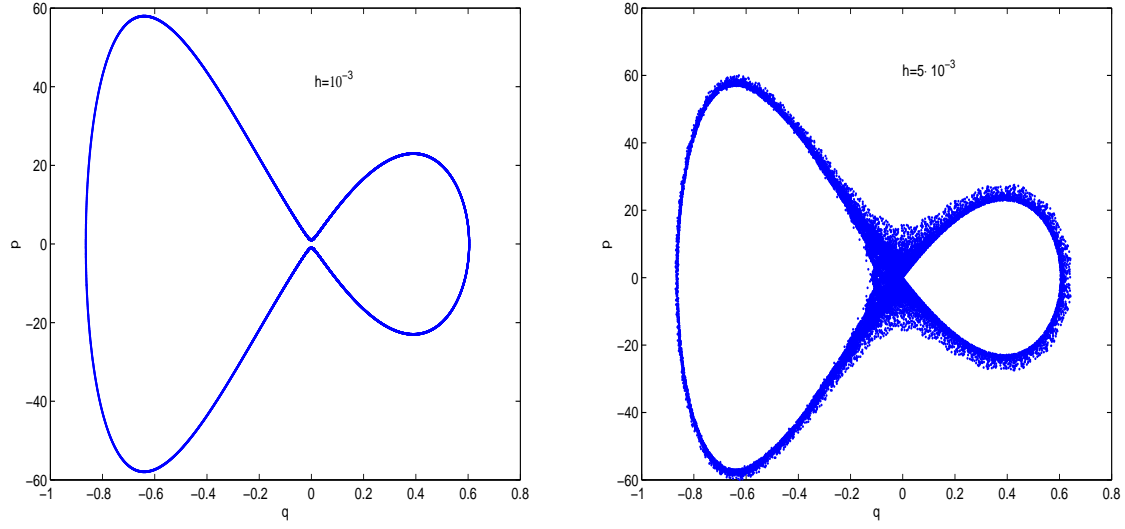


Figure 3: phase portraits for GAUSS2, $h = 10^{-3}$ (left) and $h = 5 \cdot 10^{-3}$ (right).

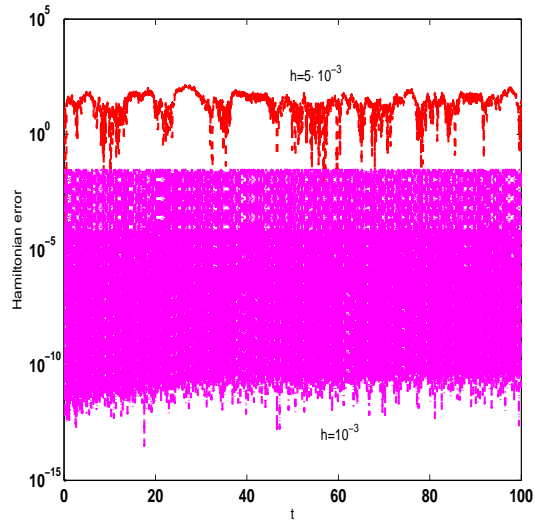


Figure 4: Hamiltonian error for GAUSS2, $h = 10^{-3}, 5 \cdot 10^{-3}$.